

19) RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

11) N° de publication : 2 766 938
(à n'utiliser que pour les commandes de reproduction)

21) N° d'enregistrement national : 98 07744

51) Int Cl⁶ : G 06 F 13/42, H 04 L 29/06

12) DEMANDE DE BREVET D'INVENTION A1

22) Date de dépôt : 19.06.98.
30) Priorité : 20.06.97 JP 16480897.

71) Demandeur(s) : SONY CORPORATION — JP.

43) Date de mise à la disposition du public de la demande : 05.02.99 Bulletin 99/05.

72) Inventeur(s) : MUTO TAKAYASU.

56) Liste des documents cités dans le rapport de recherche préliminaire : Ce dernier n'a pas été établi à la date de publication de la demande.

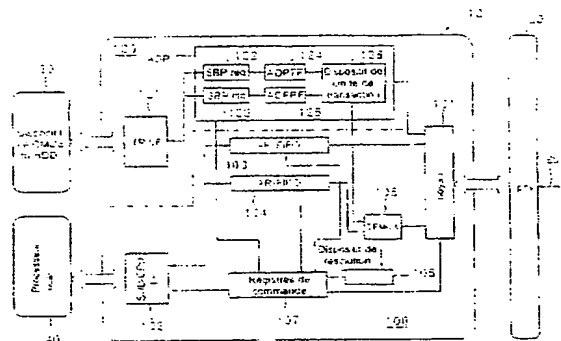
73) Titulaire(s) :

60) Références à d'autres documents nationaux apparentés :

74) Mandataire(s) : CABINET BEAU DE LOMENIE.

54) CIRCUIT D'INTERFACE SERIE ET PROCEDE DE TRAITEMENT DE SIGNAUX ASSOCIE A CE CIRCUIT.

57) Il est proposé un circuit d'interface série pouvant convertir un grand volume de données en paquets sur la base d'une norme prédéterminée d'émission et de réception. et pouvant effectuer un traitement régulier d'émission et de réception. le circuit comprenant un circuit couche de transaction (120), faisant fonction d'un circuit de traitement de données (ADP) auquel un dispositif de mémorisation (103, 104, 124, 125) est connecté, qui lit les données du dispositif de mémorisation, ajoute une étiquette de transaction auto-indicatrice, et émet les données sous la forme d'un paquet asynchrone à destination d'un bus d'interface série (BS) et, lors du transfert de données d'un autre noeud au dispositif de mémorisation, produit un paquet de demande auquel il a ajouté une étiquette auto-indicatrice et émet celui-ci à destination du bus d'interface série (BS), reçoit un paquet de réponse relatif à ce paquet de demande de la part de l'autre noeud, extrait une partie de données du paquet de réponse, et transfère celle-ci au dispositif de mémorisation.



FR 2 766 938 - A1



La présente invention concerne les circuits d'interface série numériques et, plus particulièrement, elle concerne un circuit d'interface série connecté à un dispositif de mémorisation, par exemple une mémoire sur disque dur (HDD), une mémoire morte (ROM) du type disque numérique polyvalent (DVD), une mémoire morte sur disque optique numérique (CD-ROM), ou un dispositif de sauvegarde sur bande magnétique, ainsi qu'un procédé de traitement de signaux associé.

Ces dernières années, comme interface permettant d'assurer le transfert de données multimédia, le bus série à hautes performances IEEE 1394 (le sigle IEEE signifiant Institute of Electrical and Electronic Engineers), qui permet de réaliser un transfert de données rapide et un transfert en temps réel, est devenu la norme.

Dans le transfert de données à l'aide de cette interface série IEEE 1394, l'opération de transfert effectuée dans un réseau est appelée une "sous-action". Deux sous-actions sont prescrites.

L'une est un mode de transfert asynchrone servant à faire des demandes classiques, à demander un accusé de réception et à confirmer une réception, tandis que l'autre est un mode de transfert isochrone dans lequel les données sont toujours envoyées depuis un certain noeud une fois en 125 μ s.

De cette manière, les données présentes sur une interface série IEEE 1394 ayant les deux modes de transfert sont transférées en unités de paquets. Toutefois, dans la norme IEEE 1394, la plus petite unité de données que l'on manipule est le quadruplet (soit 4 octets, ou 32 bits).

Dans la norme IEEE 1394, les données d'ordinateur sont habituellement transférées à l'aide du mode de transfert asynchrone, comme représenté sur les figures 5A et 5B.

Le mode de transfert asynchrone, comme représenté sur la figure 5A, comporte trois états de transition, à savoir l'arbitrage (arb) servant à la sécurité du bus, la transmission en paquets servant à transférer les données, et l'accusé de réception (ack).

La transmission en paquets est exécutée selon le format représenté sur la figure 5B.

Un premier quadruplet du paquet à transférer est constitué d'une région d'identification (ID) de destination à 16 bits, une région d'étiquette de transaction (tl) à 6 bits, une région de code de ré-essai (rt) à 2 bits, une région de code de transaction (tcode) à 4 bits, et une région de priorité (pri) à 4 bits.

La région ID de destination indique un numéro de bus et un numéro de noeud pour ce noeud, tandis que la région de priorité indique un niveau de priorité.

5 Un deuxième quadruplet et un troisième quadruplet sont constitués d'une région ID de source à 16 bits et d'une région de déplacement de destination à 48 bits.

La région ID de source indique l'identification du noeud qui envoie le paquet, tandis que la région de déplacement de destination est constituée par une région de niveaux hauts et bas continus et indique une adresse d'un espace d'adressage du noeud de destination.

10 Un quatrième quadruplet est constitué d'une région de longueur de données à 16 bits et d'une région de code de transaction étendu (tcode étendu) à 16 bits.

La région de longueur de données indique le nombre d'octets du paquet reçu, tandis que la région tcode étendu est une région indiquant une action de verrouillage (synchronisation) réelle effectuée par les données de ce paquet, où la région tcode indique une transaction de verrouillage.

15 Une région CRC (code de redondance cyclique) d'en-tête, ajoutée au quadruplet avant la région de la zone de données, est un code de détection d'erreur de l'en-tête du paquet.

20 De plus, la région CRC de données, ajoutée au quadruplet après la région de données (zone de données) est le code de détection d'erreur de la zone de données.

25 En outre, la figure 6 présente une vue d'un exemple de la configuration de base d'un paquet de transmission isochrone.

Comme on peut le voir sur la figure 6, dans un paquet de transmissions isochrone, le premier quadruplet est l'en-tête 1394, le deuxième quadruplet est un CRC d'en-tête, le troisième quadruplet est un en-tête 1 de CIP, le quatrième quadruplet est un en-tête 2 de CIP, le cinquième quadruplet est un en-tête de paquet de source (SPH), et le sixième quadruplet et les quadruplets suivants sont des régions de données. Le dernier quadruplet est une région CRC de données.

30 L'en-tête 1394 comprend une région de longueur de données représentant la longueur de données, une région de canal indiquant le numéro de canal à transférer par ce paquet (un nombre quelconque entre 0 et 63), une région de

code représentant le code de traitement, et un code synchrone (sy) prescrit par chaque application.

Le CRC d'en-tête est le code de détection d'erreur de l'en-tête du paquet.

5 L'en-tête 1 de CIP comprend une région d'identification de noeud de source (SID) pour le numéro du noeud de transmission, une région de taille de bloc de données (DBS) pour la longueur du bloc de données, une région de nombre de fractionnements (FN) pour le nombre de fractionnements des données dans la formation des paquets, une région de comptage de remplissage des
10 quadruplets (QPC) pour le nombre de quadruplets de données de remplissage, une région SPH pour un drapeau indiquant l'existence ou la non-existence d'un en-tête de paquet de source et une région de compteur de continuité de blocs de données (DBC) pour le compteur détectant le nombre de paquets isochrones.

On note que la région DBS représente le nombre de quadruplets
15 transférés par un seul paquet isochrone.

L'en-tête 2 de CIP comprend une région FMT pour le format de signal indiquant le type des données à transférer et une région de zone dépendant du format (FDF) qui est utilisée en correspondance avec le format des signaux.

L'en-tête SPH possède une région de consignation de temps dans
20 laquelle est positionnée une valeur obtenue par addition d'une valeur de retard fixe à un axe sur lequel le paquet du train de transport arrive.

De plus, le CRC de données est le code de détection d'erreur de la zone de données.

Comme expliqué ci-dessus, dans le transfert usuel de données
25 d'ordinateur qui est effectué dans le mode de transfert asynchrone, on utilise comme protocole le protocole SBP-2 (protocole de bus série 2).

Selon ce protocole, lorsque des données sont transférées en provenance d'un dispositif de mémorisation, c'est-à-dire une cible, à un ordinateur principal, c'est-à-dire un initiateur, le transfert s'effectue par écriture des données
30 venant du dispositif de mémorisation dans une mémoire du ordinateur principal. Lorsque des données sont transférées du ordinateur principal à la cible, le transfert est effectué par le dispositif de mémorisation lisant les données de la mémoire du ordinateur principal.

Toutefois, aucun système de circuit de traitement n'a encore été établi
35 pour commander ce que l'on pourrait appeler une "couche de transaction" permettant de convertir le grand volume de données à mémoriser dans le dispositif

de mémorisation ou à lire dans le dispositif de mémorisation, en paquets de la norme IEEE 1394 d'émission et de réception.

De plus, dans un système de circuit permettant de réaliser le mode de transfert asynchrone et le mode de transfert isochrone, il est également nécessaire de configurer le système de façon à effectuer un traitement de réception régulier en fonction du contenu des données.

Un but de l'invention est de produire un circuit d'interface série pouvant convertir un grand volume de données en paquets sur la base d'une norme prédéterminée d'émission et de réception et pouvant effectuer un traitement régulier d'émission et de réception, ainsi qu'un procédé de traitement de signaux associé.

Pour réaliser le but ci-dessus indiqué, selon un premier aspect de l'invention, il est proposé un circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud connecté à son propre noeud par l'intermédiaire d'un bus d'interface série, ayant un circuit de traitement de données permettant d'ajouter une étiquette auto-indicatrice aux données lues à destination de son propre noeud afin de produire un paquet d'émission et de le transmettre au bus d'interface série.

De préférence, le circuit d'interface série possède un circuit de commande permettant de recevoir un paquet de commande de la part d'un autre noeud et de faire démarrer le circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données, de son propre noeud à l'autre noeud.

De plus, le circuit d'interface série possède un circuit de démultiplexage servant à recevoir un paquet transféré depuis un autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est un paquet de commande, et à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse se rapportant à un paquet d'émission.

Selon un deuxième aspect de l'invention, il est proposé un circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud, connecté à son propre noeud par l'intermédiaire d'un bus d'interface série, ayant un circuit de traitement de données qui, lors du transfert des données de l'autre noeud à son propre noeud, produit un paquet de demande auquel il est ajoutée une étiquette auto-indicatrice, émet celui-ci à destination du bus d'interface série, reçoit un paquet de réponse relativement à

ce paquet de demande de la part de l'autre noeud, extrait une partie de données dans le paquet de réponse, et effectue le transfert de cette dernière.

De préférence, le circuit d'interface série possède un circuit de commande destiné à recevoir un paquet de commande de la part d'un autre noeud et à faire démarrer le circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données de l'autre noeud à son propre noeud.

De plus, le circuit d'interface série possède un circuit de démultiplexage servant à recevoir un paquet transféré depuis un autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est un paquet de commande, et à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission.

Selon un troisième aspect de l'invention, il est proposé un circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone et d'un paquet isochrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, ayant un circuit de traitement de données qui permet d'ajouter une étiquette auto-indicatrice à des données lues afin de produire un paquet d'émission et d'émettre celui-ci à destination du bus d'interface série ; un circuit de commande servant à recevoir un paquet de commande de la part de l'autre noeud et à faire démarrer le circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données de son propre noeud à l'autre noeud ; et un circuit de démultiplexage servant à recevoir un paquet transféré depuis l'autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est un paquet de commande, à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission, et à délivrer celui-ci au côté application lorsque le paquet reçu est fait de données de paquet de train isochrone.

De préférence, le circuit de démultiplexage distingue la nature d'un paquet de réponse sur la base de l'étiquette et délivre les données de paquet de train aux différents côtés application correspondants pour chaque canal.

Selon un quatrième aspect de l'invention, il est proposé un circuit d'interface série, permettant d'effectuer l'émission et la réception d'un paquet asynchrone et d'un paquet isochrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, ayant un circuit de traitement de

données qui, lors du transfert de données de l'autre noeud à son propre noeud, produit un paquet de demande auquel il a ajouté une étiquette auto-indicatrice et émet celui-ci à destination du bus d'interface série, reçoit un paquet de réponse relativement à ce paquet de demande de la part de l'autre noeud, extrait la partie de données dans le paquet de réponse, et effectue le transfert de celle-ci ; un circuit de commande servant à recevoir un paquet de commande de la part de l'autre noeud et à faire démarrer le circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données de l'autre noeud à son propre noeud ; et un circuit de démultiplexage servant à recevoir un paquet transféré depuis l'autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est le paquet de commande, à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission, et à délivrer celui-ci au côté application lorsque le paquet reçu est fait de données de paquet de train isochrone.

De préférence, le circuit de démultiplexage distingue la nature d'un paquet de réponse sur la base de l'étiquette ajoutée au paquet de réponse et du code de réponse prédéterminé et délivre les données de paquet de train aux différents côtés application correspondants pour chaque canal.

Selon un cinquième aspect de l'invention, il est proposé un circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet, auquel une information de destination de fourniture a été ajoutée, entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, ayant un circuit de démultiplexage qui permet de démultiplexer un paquet reçu sur la base de l'information de destination de fourniture du paquet reçu et de délivrer celui-ci à la destination de fourniture correspondante.

De préférence, le paquet reçu est un paquet asynchrone, et l'information de destination de fourniture est une information d'étiquette de transaction et de code de transaction.

Selon une autre possibilité, le paquet reçu est un paquet isochrone, et l'information de destination de fourniture est une information de code de transaction et de canal.

Selon un sixième aspect de l'invention, il est proposé un procédé de traitement de signaux pour un circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, comprenant,

lors du transfert de données de l'autre noeud à son propre noeud, les opérations consistant à produire un paquet de demande auquel il a été ajouté une étiquette d'auto-indication, à transmettre celui-ci au bus d'interface série, à recevoir un paquet de réponse relativement à ce paquet de demande de la part de l'autre noeud, et à extraire la partie de données dans le paquet de réponse.

De préférence, lors de la réception d'un paquet de réponse, le paquet de réception de données relatif à son propre noeud est extrait sur la base de l'étiquette ajoutée au paquet de réponse et d'un code de réponse prédéterminé.

Avec le circuit selon l'invention, dans le cas, par exemple, d'une demande de son propre noeud à un autre noeud, concernant le transfert de données du dispositif de mémorisation à l'autre noeud, les données du dispositif de mémorisation sont extraites par le circuit de traitement de données. Ensuite, une étiquette d'auto-indication est ajoutée aux données lues, qui sont ensuite transmises, sous la forme d'un paquet asynchrone d'émission, à destination du bus d'interface série. En outre, selon l'invention, le circuit de traitement de données est mis en service lorsque le circuit de commande reçoit un paquet de commande indiquant une demande de transfert de données de son propre noeud à l'autre noeud.

De plus, le circuit de démultiplexage reçoit un paquet transféré de l'autre noeud via le bus d'interface série. Lorsque le paquet reçu est un paquet de commande, il est délivré au circuit de commande, tandis que, lorsqu'il est un paquet de réponse se rapportant à un paquet d'émission, il est délivré au circuit de traitement de données.

De plus, selon le circuit de l'invention, lors du transfert de données de l'autre noeud à son propre noeud, le circuit de traitement de données produit un paquet de demande auquel il est ajouté une étiquette d'auto-indication et transmet celui-ci au bus d'interface série.

De plus, lorsqu'un paquet de réponse relatif à ce paquet de demande est reçu de la part de l'autre noeud, la partie de données est extraite du paquet de réponse et est transférée par exemple au dispositif de mémorisation.

De plus, selon le circuit de l'invention lorsque celle-ci s'applique non seulement à une communication asynchrone, mais aussi à une communication isochrone, le circuit de démultiplexage reçoit le paquet transféré depuis l'autre noeud via le bus d'interface série. Lorsque le paquet reçu est un paquet de commande, il est délivré au circuit de commande, et, lorsque le paquet reçu est un paquet de réponse relatif à un paquet d'émission, il est délivré au circuit de traite-

ment de données, tandis que, lorsque le paquet reçu est fait de données de paquet de train isochrone, il est délivré au côté application.

De plus, le circuit de démultiplexage distingue la nature d'un paquet de réponse sur la base de l'étiquette ajoutée au paquet de réponse et d'un code de réponse prédéterminé et délivre les données de paquet de train aux différents côtés application correspondants pour chaque canal respectif.

De plus, selon le circuit de l'invention, le circuit de démultiplexage distingue la nature de la destination de fourniture du paquet reçu sur la base de l'information de destination de fourniture du paquet reçu et délivre celui-ci à la destination de fourniture correspondante.

Par exemple, lorsque le paquet reçu est un paquet asynchrone, il distingue la nature de la destination de fourniture sur la base d'une information d'étiquette de transaction et de code de transaction.

De plus, lorsque le paquet reçu est un paquet isochrone, il distingue la nature de la destination de fourniture sur la base de l'information d'étiquette de transaction et de canal.

Selon le procédé de l'invention, lors du transfert de données de l'autre noeud à son propre noeud, un paquet de demande auquel il a été ajoutée une étiquette d'auto-indication est produit et est transmis au bus d'interface série. Ensuite, lorsqu'un paquet de réponse relatif à ce paquet de demande est reçu de la part de l'autre noeud, le paquet de réception de données relatif à son propre noeud est extrait sur la base de l'étiquette ajoutée au paquet de réponse et d'un code de réponse prédéterminé, et la partie de données est extraite du paquet de réponse.

La description suivante, conçue à titre d'illustration de l'invention, vise à donner une meilleure compréhension de ses caractéristiques et avantages ; elle s'appuie sur les dessins annexés, parmi lesquels :

la figure 1 est un schéma fonctionnel montrant la configuration d'un premier mode de réalisation d'un circuit d'interface série IEEE 1394 selon l'invention ;

la figure 2 est une vue illustrant de manière résumée une opération d'émission dans un circuit couche de transaction selon l'invention ;

la figure 3 est une vue illustrant, de manière résumée, une opération de réception dans un circuit couche de transaction selon l'invention ;

la figure 4 est un schéma fonctionnel montrant la configuration d'un deuxième mode de réalisation du circuit d'interface série IEEE 1394 selon l'invention ;

les figures 5A et 5B sont des vues expliquant un transfert asynchrone selon la norme IEEE 1394 ; et

la figure 6 est une vue illustrant un exemple de la configuration de base d'un paquet de transmission isochrone.

5

PREMIER MODE DE REALISATION

La figure 1 est un schéma fonctionnel montrant la configuration d'un premier mode de réalisation d'un circuit d'interface série IEEE 1394 selon l'invention.

10 On note que ce circuit d'interface série est configuré pour effectuer le transfert de données d'ordinateur manipulées dans une transmission asynchrone. Pour cette raison, sur la figure 1, la configuration concrète du circuit relatif au côté de transmission isochrone n'est pas représentée.

15 Ce circuit d'interface série est constitué par un circuit intégré formant la couche de liaison/transaction, un circuit 20 formant la couche physique, un dispositif de commande 30 d'une mémoire à disque dur (HDD) non représenté, faisant fonction du dispositif de mémorisation, et un processeur, ou dispositif de traitement, local 40 faisant fonction de l'ordinateur principal.

20 Le circuit intégré couche de liaison/transaction 10 est constitué par l'intégration d'un circuit 100 formant la couche de liaison et d'un circuit 120 formant la couche de transaction et assure la commande du transfert asynchrone et la commande du circuit couche physique 20, sous commande du processeur local 40.

25 Le circuit formant la couche de liaison 100 est constitué, comme représenté sur la figure 1, d'un noyau de liaison 101, d'un circuit d'interface de CPU (Sub-CPU I/F) 102, d'un registre FIFO d'utilisation en émission (le sigle FIFO se rapportant à un registre du type "premier entré, premier sorti" et étant désigné ici par AT-FIFO) 103 qu'on utilise pour les communications asynchrones, d'un registre FIFO d'utilisation en réception (AR-FIFO) 104 qu'on utilise pour les transmissions asynchrones, d'un circuit de démultiplexage (DEMUX) 105 servant à distinguer la nature d'un paquet reçu, d'un dispositif de résolution 106 d'utilisation en auto-identification, et d'un registre de commande (noté ici CR) 107.

35 Le noyau de liaison 101 est constitué d'un circuit d'émission destiné à un paquet d'utilisation en transmission asynchrone et à un paquet d'utilisation en transmission isochrone, par l'intermédiaire duquel des instructions et des données d'ordinateur sont transférées, d'un circuit de réception, d'un circuit d'interface

possédant le circuit couche physique 20 et servant à commander directement le bus série (BS) IEEE 1394 de ces paquets, d'un dispositif de positionnement temporel des cycles qui est repositionné toutes les 125 ms, un dispositif de contrôle de cycles, et un circuit de contrôle CRC.

5 Il effectue le traitement de transmission, ainsi que des opérations analogues, pour les données d'ordinateur lues dans le disque dur non représenté et formées en paquets de transmission prédéterminés dans le circuit couche de transaction 120.

10 On note que, sur la figure 1, comme mentionné ci-dessus, le registre FIFO et les autres éléments du côté de transmission isochrone ont été omis.

Le circuit 102 d'interface de CPU effectue l'arbitrage des opérations d'écriture, de lecture, etc., des paquets d'utilisation en transmission asynchrone entre le processeur local 40 et les registres FIFO 103 et 104 d'utilisation en émission et en réception, ainsi que l'arbitrage pour l'émission et la réception de
15 diverses données entre le processeur local 40 et le registre de commande 107.

Par exemple, il émet les instructions d'utilisation en commande du disque dur faisant fonction du dispositif de mémorisation, qui sont transmises depuis l'ordinateur principal faisant fonction de l'initiateur via le bus d'interface, BS, de la norme IEEE 1394 et sont stockées dans le registre FIFO d'utilisation en
20 réception, au processeur local 40. Les données servant à faire démarrer le circuit 120 couche de transaction pour l'émission et la réception des données d'ordinateur sont positionnées depuis le processeur local 40 dans le CR 107 via le circuit d'interface de CPU 102.

Le registre FIFO d'utilisation en émission 103 stocke un paquet
25 d'utilisation en transmission asynchrone devant être émis à destination du bus série BS de la norme IEEE 1394. Les données stockées sont fournies au noyau de liaison 101.

De plus, le registre FIFO d'utilisation en réception 104 stocke un paquet d'utilisation en transmission asynchrone qui est émis via le bus série IEEE
30 1394, par exemple, les instructions d'utilisation en commande, et autres, du disque dur faisant fonction du dispositif de mémorisation, par l'action du circuit de démultiplexage 105.

Le circuit de démultiplexage 105 contrôle le code de transaction tcode et l'étiquette de transaction t1 se trouvant dans le premier quadruplet du paquet de
35 transmission asynchrone via le noyau de liaison 101, détermine s'il s'agit d'un paquet de réponse venant de l'initiateur, c'est-à-dire de l'ordinateur principal, pour

aller à la cible, c'est-à-dire le circuit couche de transaction, ou d'un autre paquet, n'introduit que le paquet de réponse dans le circuit couche de transaction 102, et stocke les autres paquets dans le registre FIFO d'utilisation en réception 104.

5 On note que l'étiquette de transaction t1 utilisée pour le contrôle de détermination est ordinairement positionnée à "a". Au titre du tcode (code de transaction), des données différentes sont positionnées en fonction du fait qu'il s'agit d'une demande d'écriture et d'une réponse ou bien d'une demande de lecture et d'une réponse.

10 Plus spécialement, le tcode est positionné à "0" dans le cas d'une demande d'écriture et d'une écriture de quadruplet, tandis qu'il est positionné à "1" dans le cas d'une écriture de bloc.

De plus, il est positionné à "2" dans le cas d'une réponse d'écriture. Dans le cas d'une demande de lecture et d'une lecture de quadruplet, il est positionné à "4", et, dans le cas d'une lecture de bloc, il est positionné à "5".

15 De plus, dans le cas d'une réponse de lecture, il est positionné à "6/7".

Le dispositif de résolution 106 analyse le paquet à auto-identification transmis via le bus série BS de la norme IEEE 1394 et stocke celui-ci dans le CR 107. De plus, il a également pour fonctions de contrôler les erreurs, de compter le nombre de noeuds, etc.

20 Le circuit couche de transaction 120 a pour fonction d'effectuer automatiquement l'émission et la réception des données de périphériques d'ordinateur (mémoires à disques durs dans le présent mode de réalisation) sous forme de paquets asynchrones sur la base de la norme SBP-2 (protocole de bus série 2).

25 De plus, le circuit couche de transaction 120 est doté d'une fonction de ré-essai et d'une fonction de détection de fin de temps imparti séparé.

30 La fonction de ré-essai est une fonction amenant à ré-émettre le paquet de demande correspondant lorsqu'un code d'accusé de réception "occupé*" est renvoyé après l'émission d'un paquet de demande. Lorsque le paquet est ré-émis, l'émission s'effectue après positionnement, de "00" à "01", de la région rt à 2 bits, qui est placée dans le premier quadruplet du paquet d'émission.

La fonction de détection de fin de temps imparti séparé est une fonction permettant de déterminer lorsque le temps prévu pour le retour du paquet de réponse est arrivé à épuisement.

35 Ce circuit couche de transaction 120 est constitué d'un circuit d'interface de données de transport (TP I/F) 121, d'un circuit générateur de paquet de demande (SBPreq) 122, d'un circuit de décodage de paquet de réponse

(SBPrsp) 123, d'un registre FIFO d'utilisation en demande (FIFO de demande : ADPTF) 124, d'un registre FIFO d'utilisation en réponse (FIFO de réponse : ADPRF) 125 et d'un dispositif de commande de transaction 126.

5 Un circuit de traitement de données ADP est constitué du circuit générateur de paquet de demande (SBPreq) 122, du circuit de décodage de paquet de réponse (SBRrsp) 123, du registre FIFO d'utilisation en demande (FIFO de demande : ADPTF) 124, du registre FIFO d'utilisation en réponse (FIFO de réponse : ADPRF) 125, et du dispositif de commande de transaction 126.

10 Le circuit 121 d'interface de données de transport effectue l'arbitrage pour l'émission et la réception de données entre le dispositif 30 de commande du HDD et les circuits 122 générateur de paquet de demande et 123 décodeur de paquet de réponse.

15 Lors de la réception d'une instruction de démarrage de transfert de données, provenant du CR 107 du circuit couche de liaison 100, dans le cas de l'émission (écriture), le circuit 122 générateur de paquet de demande divise les données d'ordinateur enregistrées sur un disque dur, non représenté, qui sont obtenues via le circuit 121 d'interface de données de transport, selon la norme SBP-2 en un ou plusieurs blocs de données, de façon qu'elles puissent être divisées en paquets, ajoute un en-tête 1394 indiquant l'étiquette de transaction t1
20 (= a) et les stocke dans le registre FIFO d'utilisation en demande 124.

De plus, dans le cas de la réception (lecture), il forme l'instruction de demande de lecture de bloc 1394 ayant l'adresse désignée et la longueur de données désignée en un paquet en désignant une ou plusieurs étiquettes de transaction t1 (= a), ou analogue, selon la norme SBP-2 et les stocke dans le
25 registre FIFO d'utilisation en demande 124.

Le circuit 123 décodeur de paquet de réponse lit les données du paquet de réponse stockées dans le registre FIFO d'utilisation en réponse 125 au moment de la réception, retire du paquet de réponse l'en-tête 1394, et délivre les données lues, via le circuit d'interface de données de transport 121, au dispositif 30 de
30 commande du HDD à un moment prédéterminé.

Le registre FIFO d'utilisation en demande 124, dans le cas de l'émission (écriture), stocke les données d'émission divisées en paquets et, dans le cas de la réception (lecture), stocke l'instruction de demande de lecture de blocs 1394.

Le registre FIFO d'utilisation en réponse 125 stocke les données de réception émises depuis le côté de l'ordinateur principal via le bus série BS de la norme 1394.

5 Le dispositif de commande de transaction 126 lit les données d'émission divisées en paquets stockées dans le registre FIFO d'utilisation en demande 124 au moment de l'émission et la commande de demande de lecture de blocs 1394 stockée dans le registre FIFO d'utilisation en demande 124 au moment de la réception et il commande le signal de sortie à destination du noyau de liaison 101 du circuit couche de liaison 100.

10 De plus, au moment de l'émission, lors de la réception d'un paquet de réponse en provenance du circuit de démultiplexage 105 du circuit couche de liaison 100, son code de ré-essai rcode est écrit dans le CR 107. Au moment de la réception l'en-tête 1394 est retiré du paquet de réponse venant du circuit de démultiplexage 105, et les données du paquet sont stockées dans le registre FIFO
15 d'utilisation en réponse 125.

On va maintenant présenter des explications au sujet des opérations d'émission et de réception des données d'ordinateur lorsque des paquets déterminés selon la norme SBP-2 sont transférés dans la configuration ci-dessus présentée.

20 Tout d'abord, on explique l'opération d'émission, c'est-à-dire le cas où les données sont transférées de la cible, à savoir le disque dur, à l'initiateur, c'est-à-dire l'ordinateur principal, et lorsque l'opération d'écriture des données dans la mémoire de l'ordinateur principal, en provenance du dispositif de mémorisation (disque dur) est effectué.

25 Les données du paquet, comme par exemple le bloc de demande d'opération (ORB) basé sur la norme SBP-2, transférées depuis l'ordinateur principal via le bus série BS de la norme 1394 est appliqué en entrée au circuit de démultiplexage 105 via le circuit couche physique 20 et le noyau de liaison 101 du circuit couche de liaison 100.

30 Le circuit de démultiplexage 105, dès réception du paquet reçu, détermine si le paquet est un paquet de réponse relativement à la cible, c'est-à-dire le circuit couche de transaction, en provenance de l'ordinateur principal, ou un autre paquet.

35 Alors, dans ce cas, s'il s'agit d'un autre paquet, les données reçues sont stockées dans le registre FIFO d'utilisation en réception 104. Les données reçues, comme par exemple l'ORB, stockées dans le registre FIFO d'utilisation en

réception 105 sont appliquées en entrée au processeur local 40 via le circuit d'interface de CPU 102.

Le processeur local 40 initialise le registre d'utilisation en circuit couche de transaction du CR 107 en fonction du contenu de l'ORB, via le circuit
5 d'interface de CPU 102.

Alors, le circuit couche de transaction 120 commence son fonctionnement.

Dans le circuit couche de transaction 120 qui vient de démarrer, le circuit 122 générateur de paquet de demande commence à demander des données
10 au dispositif 30 de commande du HDD via le circuit d'interface de données de transport (TP I/F) 121.

Selon la demande, les données d'émission envoyées via le circuit d'interface de données de transport 121 sont divisées en un ou plusieurs blocs de données. de façon à pouvoir être divisées en paquets selon la norme SBP-2 dans le
15 circuit générateur de paquet de demande 122. Un en-tête 1394 ayant une étiquette de transaction t1 (= a) désignée, c'est-à-dire d'indication ou analogue, est ajouté, et le résultat est automatiquement stocké dans le registre FIFO d'utilisation en demande 124.

Lorsque des données de la taille d'un paquet 1394, ou plus, sont
20 stockées dans le registre FIFO d'utilisation en demande 124, les données sont envoyées au noyau de liaison 101 du circuit couche de liaison 100 par le dispositif de commande de transaction 126.

Ensuite, le noyau de liaison 101 applique un arbitrage au bus série BS 1394 via le circuit couche physique 20.

De ce fait, si le bus peut être dans des conditions de sécurité, un
25 paquet de demande d'écriture contenant les données de transfert est émis à destination de l'ordinateur principal via le circuit couche physique 20 et le bus série BS 1394.

Après l'émission, un code d'accusé de réception (Ack) relatif au paquet
30 de demande d'écriture et, dans certain cas, au paquet de réponse d'écriture est envoyé depuis l'ordinateur principal et est appliqué en entrée au circuit de démultiplexage 105 via le circuit couche physique 20 et le noyau de liaison 101 du circuit couche de liaison 100.

Dans le circuit de démultiplexage 105, un contrôle du code de
35 transaction tcode et de l'étiquette de transaction t1 du paquet reçu est effectué, et, lorsqu'il a été déterminé que le paquet reçu est le paquet de réponse relatif au

circuit couche de transaction faisant fonction de la cible, en provenance de l'ordinateur principal, le paquet de réponse est appliqué en entrée au dispositif de commande de transaction 126 du circuit couche de transaction 120.

5 Le dispositif de commande de transaction 126 transmet les données suivantes au noyau de liaison 101 si le code Ack et le code de réponse du paquet de réponse d'entrée sont normaux. L'opération ci-dessus se répète pour écrire (émettre) les données d'ordinateur à destination de la mémoire de l'ordinateur principal.

10 Une représentation du fonctionnement du circuit couche de transaction 120 pour l'émission ci-dessus décrite est présentée sur la figure 2.

On va ensuite donner des explications concernant l'opération de réception, c'est-à-dire un cas où des données sont transférées de l'ordinateur principal à la cible et où le dispositif de mémorisation (disque dur) effectue l'opération de lecture des données dans la mémoire de l'ordinateur principal.

15 Les données du paquet, comme par exemple l'ORB, selon la norme SBP-2, qui ont été transférées en provenance de l'ordinateur principal via le bus série BS 1394, sont appliquées en entrée au circuit de démultiplexage 105 via le circuit couche physique 20 et le noyau de liaison 101 du circuit couche de liaison 100.

20 Le circuit de démultiplexage 105 reçoit le paquet reçu et détermine si le paquet reçu est un paquet de réponse relativement à la cible, c'est-à-dire au circuit couche de transaction, provenant de l'ordinateur principal, ou bien un autre paquet.

25 Alors, dans ce cas, puisque le paquet reçu est un autre paquet, les données reçues sont stockées dans le registre FIFO d'utilisation en réception 104.

Les données reçues, comme par exemple l'ORB, stockées dans le registre FIFO d'utilisation en réception 104, sont appliquées en entrée au processeur local 40 via le circuit d'interface de CPU 102.

30 Le processeur local 40 initialise le registre d'utilisation en circuit couche de transaction du CR 107 en fonction du contenu de l'ORB, via le circuit d'interface de CPU 102.

De cette manière, le circuit couche de transaction 120 fait commencer son fonctionnement.

35 Dans le circuit couche de transaction 120 activé, le circuit 122 générateur de paquet de demande forme l'instruction de demande de lecture de blocs 1394 ayant l'adresse désignée et la longueur de données désignée en un

paquet obéissant à la norme SBP-2 et stocke celui-ci dans le registre FIFO d'utilisation en demande 124.

5 Le paquet d'instruction de demande de lecture stocké dans le registre FIFO d'utilisation en demande 124 est envoyé au noyau de liaison 101 du circuit couche de liaison 100 par le dispositif de commande de transaction 126.

Ensuite, le noyau de liaison 101 applique un arbitrage au bus série BS 1394 via le circuit couche physique 20.

10 De ce fait, lorsque le bus peut être dans des conditions de sécurité, le paquet de demande de lecture est émis à destination de l'ordinateur principal via le circuit couche physique 20 et le bus série 1394 BS.

Après l'émission, un code Ack relatif au paquet de demande de lecture et au paquet de réponse de lecture contenant les données qui ont la longueur désignée est envoyé par l'ordinateur principal et appliqué en entrée au circuit de démultiplexage 105 via le circuit couche physique 20 et le noyau de liaison 101 du circuit couche de liaison 100.

15 Le circuit de démultiplexage 105 contrôle le code de transaction tcode et l'étiquette de transaction t1 du paquet reçu. Lorsqu'il détermine qu'un paquet reçu est un paquet de réponse relativement à la cible, c'est-à-dire le circuit couche de transaction, en provenance de l'ordinateur principal, il applique le paquet de réponse en entrée au dispositif de commande de transaction 126 du circuit couche de transaction 120.

20 Le dispositif de commande de transaction 126 stocke les données du paquet de réponse venant du circuit de démultiplexage 105 dans le registre FIFO d'utilisation en réponse 125.

25 Les données stockées dans le registre FIFO d'utilisation en réponse 125 sont lues par le circuit 123 décodeur de paquet de réponse, l'en-tête 1394 est retiré, et les données résultantes sont délivrées au dispositif 30 de commande du HDD via le circuit d'interface de données de transport 121 à un moment prédéterminé.

30 L'opération ci-dessus se répète pour écrire les données d'ordinateur dans le dispositif de mémorisation (disque dur), ou pour en recevoir.

Un schéma illustrant le fonctionnement du circuit couche de transaction 120 pour l'opération de réception ci-dessus décrit est présenté sur la figure 3.

35 Comme expliqué ci-dessus, selon le premier mode de réalisation, puisqu'il est prévu un circuit couche de transaction 120 faisant fonction d'un

circuit de traitement de données auquel un dispositif de mémorisation est connecté, qui lit les données du dispositif de mémorisation, ajoute une étiquette de transaction d'auto-indication, et émet les données sous la forme d'un paquet de transmission asynchrone à destination du bus d'interface série BS et, lors du
5 transfert de données d'un autre noeud au dispositif de mémorisation, produit un paquet de demande auquel il a été ajouté une étiquette d'auto-indication et émet celui-ci à destination du bus d'interface série BS, reçoit un paquet de réponse relatif à ce paquet de demande de la part de l'autre noeud, extrait une partie de données du paquet de réponse, et transfère celle-ci au dispositif de mémorisation,
10 il est possible d'émettre et de recevoir un grand volume de données, stockées dans le dispositif de mémorisation ou lues dans le dispositif de mémorisation, en paquets IEEE 1394 sur la base de la norme SBP-2 et il est possible de réaliser un transfert de données de grand volume en utilisant des paquets asynchrones pour l'interface de bus série IEEE 1394.

15 De plus, il est possible de simplifier la séquence basée sur la norme SBP-2 d'extraction de l'ORB, de transfert des données et d'émission de l'état à destination de l'initiateur, et une conception optimale devient possible lors de la connexion d'une mémoire à disque, d'un dispositif de sauvegarde sur bande magnétique, ou d'autres périphériques d'ordinateur, à un bus série IEEE 1394.

20 De plus, puisque le registre FIFO d'utilisation en demande 124 et le registre FIFO d'utilisation en réponse 125 sont prévus dans le circuit couche de transaction 120 et que le registre FIFO d'utilisation en émission 103 et le registre FIFO d'utilisation en réception 104 sont prévus dans le circuit couche de liaison
25 100, l'émission et la réception de paquets 1394 ordinaires autres que des données peuvent s'effectuer en parallèle avec le transfert de données par le registre FIFO d'utilisation en demande 124 et le registre FIFO d'utilisation en réponse 125.

De plus, puisqu'il est prévu le circuit de démultiplexage 105, qui contrôle le code de transaction tcode et l'étiquette de transaction t1 placés dans le premier quadruplet du paquet de transmission asynchrone via le noyau de liaison
30 101, détermine si le paquet de réponse est un paquet de réponse relativement à la cible, à savoir le circuit couche de transaction, en provenance de l'initiateur, à savoir l'ordinateur principal, ou un autre paquet, n'applique que le paquet de réponse au circuit couche de transaction 120, et stocke les autres paquets dans le registre FIFO d'utilisation en réception 104, alors, même si une erreur cruciale se
35 produit par exemple du côté du circuit couche de transaction 120 et que l'opération de lecture/écriture de données s'arrête, l'invention offre l'avantage selon lequel la

lecture des instructions introduites après les données ne devient pas impossible et que, par conséquent, des instructions peuvent être reçues de manière régulière indépendamment de l'état de l'opération de lecture/écriture des données.

5

DEUXIEME MODE DE REALISATION

La figure 4 est un schéma fonctionnel montrant la configuration d'une deuxième mode de réalisation du circuit d'interface série IEEE 1394 selon l'invention.

10 Sur la figure 4, 10a désigne un circuit intégré couche de liaison/transaction, 20 un circuit couche physique, 30 un dispositif de commande de HDD, 40 un processeur local, et 50a, 50b désignent des dispositifs de transport MPEG (Moving Picture Experts Group, norme utilisée pour le traitement d'images dans les ordinateurs multimédia).

15 Le circuit intégré couche de liaison/transaction 10a est constitué du circuit couche de liaison 100a et du circuit couche de transaction 120.

20 Le circuit couche de liaison 100a est constitué d'un noyau de liaison 101, d'un circuit d'interface de CPU 102, d'un registre FIFO d'utilisation en transmission asynchrone 103, d'un registre FIFO d'utilisation en réception asynchrone 104, d'un circuit de démultiplexage 105a, d'un dispositif de résolution 106, d'un registre de commande CR 107, de circuits d'interface d'application 108a et 108b, d'un registre FIFO d'utilisation en émission isosynchrone (IT-FIFO) 109, et de registres FIFO d'utilisation en réception isosynchrone (IF-FIFO) 110a et 110b.

25 De plus, le circuit couche de transaction 120 est constitué, comme pour la figure 1, du circuit d'interface de données de transport (TP I/F) 121, du circuit générateur de paquet de demande (SBPreq) 122, du circuit de décodage de paquet de réponse (SBPrsp) 123, du registre FIFO d'utilisation en demande (FIFO de demande : ADPTF) 124, du circuit FIFO d'utilisation en réponse (FIFO de réponse : ADPRF) 125, et du dispositif de commande de transaction 126.

30 La différence entre le deuxième mode de réalisation et le premier mode de réalisation réside en ce que le circuit couche de liaison/transaction manipule non seulement des données du mode de transmission asynchrone et des paquets 1394 usuels, mais aussi des données d'utilisation en transmission isosynchrone.

35 Plus spécialement, le circuit de démultiplexage 105a contenu dans le circuit couche de liaison 100a du circuit couche de liaison/transaction 10a,

contrôle, de la même manière que dans le cas de la figure 1, le code de transaction tcode et l'étiquette de transaction t1 du paquet de transmission asynchrone via le noyau de liaison 101, déterminé si un paquet reçu est un paquet de réponse relativement à la cible, c'est-à-dire le circuit couche de transaction, venant de l'initiateur, c'est-à-dire l'ordinateur principal, ou un autre paquet, n'applique en entrée que le paquet de réponse au circuit couche de transaction 120 et stocke les autres paquets dans le registre FIFO d'utilisation en réception 104. Lors de la réception d'un paquet de transmission isochrone, il décode la région tcode et la région de canal indiquant le numéro du canal (un nombre quelconque entre 0 et 63), servant au transfert du paquet, dans l'information d'en-tête du paquet 1394 et stocke sélectivement celui-ci dans les registres FIFO d'utilisation en réception 108a et 108b prévus en correspondance avec les canaux.

De plus, le circuit d'interface d'application (AP I/F) 108a effectue l'arbitrage de l'émission et de la réception des données du train de transport MPEG contenant le signal d'horloge, le signal de commande, etc., entre le dispositif de transport MPEG 50a et les registres FIFO d'utilisation en émission 108 et d'utilisation en réception 110a.

Le circuit d'interface d'application 108b effectue l'arbitrage pour l'émission et la réception des données de train de transport MPEG contenant le signal d'horloge, le signal de commande, etc., entre le dispositif de transport MPEG 50b et les registres FIFO d'utilisation en émission 109 et d'utilisation en réception 110b.

Avec la configuration ci-dessus, lorsque, par exemple, un paquet d'utilisation en transmission isochrone est reçu, le circuit de démultiplexage 105a décode la région du tcode et la région de canal indiquant le numéro du canal (un nombre quelconque entre 0 et 63), servant au transfert du paquet, dans l'information d'en-tête du paquet 1394 et stocke sélectivement celui-ci dans les FIFO d'utilisation en réception 108a et 108b prévus en correspondance avec les canaux.

Ensuite, l'émission et la réception des données de train de transport MPEG associées au dispositif de transport MPEG 50a ou 50b sont effectuées via le circuit d'interface d'application 108a ou 108b.

Dans le deuxième mode de réalisation également, on peut obtenir les effets analogues à ceux du premier mode de réalisation.

On note que, dans les modes de réalisation ci-dessus, on a donné les explications en prenant comme exemple une structure de circuit dans laquelle les circuits de démultiplexage 105 et 105a étaient placés entre le noyau de liaison 101

et les registres FIFO d'utilisation en réception 104, 110a et 110b des circuits couche de liaison 100 et 100a, mais il va sans dire que l'invention peut également être appliquée à une configuration de circuit dans laquelle ils sont placés du côté de sortie de données du circuit d'interface, du registre FIFO du côté réception par exemple.

Comme expliqué ci-dessus, selon l'invention, un circuit d'interface série pouvant convertir un grand volume de données en paquets selon une norme prédéterminée d'émission et de réception et pouvant effectuer un traitement régulier de l'émission et de la réception peut être réalisé.

Bien entendu, l'homme de l'art sera en mesure d'imaginer, à partir des circuits et des procédés dont la description vient d'être donnée à titre simplement illustratif et nullement limitatif, diverses variantes et modifications ne sortant pas du cadre de l'invention.

REVENDEICATIONS

- 5 1. Circuit d'interface série servant à effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, caractérisé en ce qu'il comprend :
- un circuit de traitement de données (120, ADP) permettant d'ajouter une étiquette auto-indicatrice aux données lues à destination de son propre noeud pour produire un paquet de transmission, et d'émettre celui-ci à destination du bus d'interface série (BS).
- 10 2. Circuit d'interface série selon la revendication 1, caractérisé en ce que ledit circuit de traitement de données (120, ADP) divise les données lues en une pluralité de blocs de données, ajoute l'étiquette à chaque bloc de données, et émet ceux-ci sous la forme de paquets asynchrones d'émission à destination dudit bus d'interface série (BS).
- 15 3. Circuit d'interface série selon la revendication 2, caractérisé en ce que ledit circuit de traitement de données (120, ADP) reçoit un paquet de réponse relativement à un paquet d'émission de la part dudit autre noeud et, lorsqu'il est normal, émet le paquet d'émission suivant à destination dudit bus d'interface série (BS).
- 20 4. Circuit d'interface série selon la revendication 1, 2 ou 3, caractérisé en ce que ledit circuit de traitement de données (120, ADP) comprend un moyen de mémorisation (103, 104, 124, 125), mémorise dans ledit moyen de mémorisation le paquet d'émission produit, et émet le paquet d'émission mémorisé à destination du bus d'interface série (BS) à un moment prédéterminé.
- 25 5. Circuit d'interface série selon la revendication 1, 2, 3 ou 4, caractérisé en ce qu'il possède en outre un circuit de commande (107) destiné à recevoir un paquet de la part d'un autre noeud et à faire démarrer le fonctionnement du circuit de traitement de données (120, ADP) lorsque le contenu du paquet de commande indique une demande de transfert de données, de son propre noeud
- 30 à l'autre noeud.
6. Circuit d'interface série selon la revendication 5, caractérisé en ce qu'il comprend en outre un circuit de démultiplexage (105) servant à recevoir un paquet transféré depuis un autre noeud via le bus d'interface série (BS), à délivrer celui-ci au circuit de commande (107) lorsque le paquet reçu est un paquet de
- 35 commande, et à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission.

7. Circuit d'interface série selon la revendication 6, caractérisé en ce qu'il comprend en outre un moyen (104) de mémorisation d'utilisation en réception qui est placé entre ledit circuit de démultiplexage (105) et ledit circuit de commande (107), ledit circuit de démultiplexage mémorisant le paquet de commande démultiplexé dans ledit moyen de mémorisation d'utilisation en réception, et ledit circuit de commande lisant le paquet de commande mémorisé dans ledit moyen de mémorisation d'utilisation en réception.

8. Circuit d'interface série selon la revendication 7, caractérisé en ce qu'un moyen (103) de mémorisation d'utilisation en émission est connecté audit circuit de commande (107), ledit circuit de commande mémorise le paquet de commande dans ledit moyen de mémorisation d'utilisation en émission, et en ce qu'il comprend en outre un circuit (101) permettant d'émettre un paquet de commande d'utilisation en émission mémorisé dans ledit moyen de mémorisation d'utilisation en émission à destination dudit bus d'interface série à un moment prédéterminé.

9. Circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série (BS), caractérisé en ce qu'il comprend :

un circuit de traitement de données (120, ADP) qui, lors du transfert des données de l'autre noeud à son propre noeud, produit un paquet de demande auquel il a ajouté une étiquette auto-indicatrice, émet celui-ci à destination du bus d'interface série (BS), reçoit un paquet de réponse relativement à ce paquet de demande de la part de l'autre noeud, extrait une partie de données du paquet de réponse, et transfère celle-ci.

10. Circuit d'interface série selon la revendication 9, caractérisé en ce que ledit circuit de traitement de données (120, ADP) produit une pluralité de paquets de demande ayant des étiquettes indicatrices et émet ceux-ci à destination du bus d'interface série (BS) afin de permettre le transfert de données en paquets.

11. Circuit d'interface série selon la revendication 10, caractérisé en ce que ledit circuit de traitement de données (120, ADP) reçoit un paquet de réponse relativement à un paquet d'émission de la part dudit autre noeud et, lorsqu'il est normal, émet le paquet d'émission suivant à destination dudit bus d'interface série (BS).

12. Circuit d'interface série selon la revendication 9, caractérisé en ce que ledit circuit de traitement de données (120, ADP) comprend un premier

moyen de mémorisation (103, 124) et un deuxième moyen de mémorisation (104, 125), mémorise dans le premier moyen de mémorisation le paquet de demande produit, émet le paquet de demande mémorisé à destination du bus d'interface série (BS) à un moment prédéterminé, mémorise les données reçues dans le
5 deuxième moyen de mémorisation, et transfère les données reçues mémorisées à un moment prédéterminé.

13. Circuit d'interface série selon la revendication 10 ou 11, caractérisé en ce que ledit circuit de traitement de données (120, ADP) comprend un premier moyen de mémorisation (103, 124) et un deuxième moyen de mémorisation (104,
10 125), mémorise dans le premier moyen de mémorisation le paquet d'émission produit, émet le paquet d'émission mémorisé à destination du bus d'interface série (BS) à un moment prédéterminé, mémorise les données reçues dans le deuxième moyen de mémorisation, et transfère les données reçues mémorisées à un moment prédéterminé.

15 14. Circuit d'interface série selon la revendication 9, 10, 11 ou 13, caractérisé en ce qu'il comprend en outre un circuit de commande (107) servant à recevoir un paquet de commande de la part d'un autre noeud et à faire démarrer le fonctionnement du circuit de traitement de données (120, ADP) lorsque le contenu du paquet de commande indique une demande de transfert de données de l'autre
20 noeud à son propre noeud.

15. Circuit d'interface série selon la revendication 14, caractérisé en ce qu'il comprend en outre un circuit de démultiplexage (105) servant à recevoir un paquet transféré depuis un autre noeud via le bus d'interface série (BS), à délivrer celui-ci au circuit de commande (107) lorsque le paquet reçu est un paquet de
25 commande, et à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission.

16. Circuit d'interface série selon la revendication 15, caractérisé en ce qu'il comprend en outre un moyen (104) de mémorisation d'utilisation en réception qui est placé entre ledit circuit de démultiplexage (105) et ledit circuit de
30 commande (107), ledit circuit de démultiplexage mémorisant le paquet de commande démultiplexé dans ledit moyen de mémorisation d'utilisation en réception, et ledit circuit de commande lisant le paquet de commande mémorisé dans ledit moyen de mémorisation d'utilisation en réception.

17. Circuit d'interface série selon la revendication 16, caractérisé en ce qu'un moyen (103) de mémorisation d'utilisation en émission est connecté audit
35 circuit de commande (107), ledit circuit de commande mémorise le paquet de

commande dans ledit moyen de mémorisation d'utilisation en émission, et en ce qu'il comprend en outre un circuit (101) permettant d'émettre un paquet de commande d'utilisation en émission mémorisé dans ledit moyen de mémorisation d'utilisation en émission à destination dudit bus d'interface série à un moment prédéterminé.

18. Circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, caractérisé en ce qu'il comprend :

10 un circuit de traitement de données (120, ADP) permettant d'ajouter une étiquette auto-indicatrice à des données lues pour produire un paquet d'émission, et d'émettre celui-ci à destination du bus d'interface série (BS) et, lors du transfert de données de l'autre noeud à son propre noeud, de produire un paquet de demande auquel il a ajouté une étiquette auto-indicatrice, d'émettre celui-ci à destination du bus d'interface série, de recevoir un paquet de réponse relativement au paquet de demande de la part de l'autre noeud, d'extraire la partie de données du paquet de réponse, et de transférer celle-ci.

19. Circuit d'interface série selon la revendication 18, caractérisé en ce que ledit circuit de traitement de données (120, ADP) divise les données lues en une pluralité de blocs de données, ajoute l'étiquette ci-dessus indiquée à chaque bloc de données et émet ceux-ci sous la forme de paquets asynchrones d'émission à destination du bus d'interface série (BS), et, lors du transfert de données de l'autre noeud à son propre noeud, produit une pluralité de paquets de demande ayant des étiquettes indicatrices et émet ceux-ci à destination du bus d'interface série afin de permettre le transfert des données en paquets.

20. Circuit d'interface série selon la revendication 19, caractérisé en ce que ledit circuit de traitement de données (120, ADP) reçoit un paquet de réponse relativement à un paquet d'émission de la part dudit autre noeud et, lorsqu'il est normal, émet le paquet d'émission suivant à destination dudit bus d'interface série (BS).

21. Circuit d'interface série selon la revendication 18, 19 ou 20, caractérisé en ce que ledit circuit de traitement de données (120, ADP) comprend un premier moyen de mémorisation (103, 124) et un deuxième moyen de mémorisation (104, 125), mémorise dans le premier moyen de mémorisation le paquet d'émission produit, émet le paquet d'émission mémorisé à destination du bus d'interface série (BS) à un moment prédéterminé, mémorise les données reçues

2
dans le deuxième moyen de mémorisation, et transfère les données reçues mémorisées à un moment prédéterminé.

5 22. Circuit d'interface série selon la revendication 18, 19, 20 ou 21, caractérisé en ce qu'il comprend un circuit de commande (107) servant à recevoir un paquet de commande de la part d'un autre noeud et à faire démarrer le fonctionnement du circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données de son propre noeud à l'autre noeud et lorsqu'il indique une demande de transfert de données de l'autre noeud à son propre noeud.

10 23. Circuit d'interface série selon la revendication 22, caractérisé en ce qu'il comprend en outre un circuit de démultiplexage (105) servant à recevoir un paquet transféré depuis un autre noeud via le bus d'interface série (BS), à délivrer celui-ci au circuit de commande (107) lorsque le paquet reçu est un paquet de commande, et à délivrer celui-ci au circuit de traitement de données lorsque le
15 paquet reçu est un paquet de réponse relativement à un paquet d'émission.

24. Circuit d'interface série selon la revendication 23, caractérisé en ce qu'il comprend en outre un moyen (104) de mémorisation d'utilisation en réception qui est placé entre ledit circuit de démultiplexage (105) et ledit circuit de commande (107), ledit circuit de démultiplexage mémorisant le paquet de
20 commande démultiplexé dans ledit moyen de mémorisation d'utilisation en réception, et ledit circuit de commande lisant le paquet de commande mémorisé dans ledit moyen de mémorisation d'utilisation en réception.

25. Circuit d'interface série selon la revendication 24, caractérisé en ce qu'un moyen (103) de mémorisation d'utilisation en émission est connecté audit
25 circuit de commande (107), ledit circuit de commande mémorise le paquet de commande dans ledit moyen de mémorisation d'utilisation en émission, et en ce qu'il comprend en outre un circuit (101) permettant d'émettre un paquet de commande d'utilisation en émission mémorisé dans ledit moyen de mémorisation d'utilisation en émission à destination dudit bus d'interface série à un moment
30 prédéterminé.

26. Circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone et d'un paquet isochrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, caractérisé en ce qu'il comprend :

un circuit de traitement de données (120, ADP) permettant d'ajouter une étiquette auto-indicatrice à des données lues pour produire un paquet d'émission, et d'émettre celui-ci à destination du bus d'interface série (BS) ;

5 un circuit de commande (107) servant à recevoir un paquet de commande de la part de l'autre noeud et à faire démarrer le fonctionnement du circuit de traitement de données lorsque le contenu du paquet de demande indique une demande de transfert de données de son propre noeud à l'autre noeud ; et

10 un circuit de démultiplexage (105 ; 105a) servant à recevoir un paquet transféré depuis l'autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est un paquet de commande, à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission, et à délivrer celui-ci au côté application lorsque le paquet reçu est fait de données de paquet de train isochrone.

15 27. Circuit d'interface série selon la revendication 26, caractérisé en ce que le circuit de démultiplexage (105 ; 105a) distingue la nature d'un paquet de réponse sur la base de l'étiquette et délivre les données de paquet de train aux différents côtés application correspondant pour chaque canal.

20 28. Circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone et d'un paquet isochrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, caractérisé en ce qu'il comprend :

25 un circuit de traitement de données (120, ADP) qui, lors du transfert de données de l'autre noeud à son propre noeud, produit un paquet de demande auquel il a ajouté une étiquette auto-indicatrice et, ayant émis celui-ci à destination dudit bus d'interface série, reçoit un paquet de réponse relativement au paquet de demande en provenance de l'autre noeud, extrait la partie de données du paquet de réponse, et transfère celle-ci ;

30 un circuit de commande (107) servant à recevoir un paquet de commande de la part de l'autre noeud et à faire démarrer le fonctionnement du circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données de l'autre noeud à son propre noeud ; et

35 un circuit de démultiplexage (105 ; 105a) servant à recevoir un paquet transféré depuis l'autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est un paquet de commande, à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de

réponse relativement à un paquet d'émission, et à délivrer celui-ci au côté application lorsque le paquet reçu est fait de données de paquet de train isochrone.

5 29. Circuit d'interface série selon la revendication 28, caractérisé en ce que le circuit de démultiplexage (105 ; 105a) distingue la nature d'un paquet de réponse sur la base de l'étiquette ajoutée au paquet de réponse et d'un code de réponse prédéterminé et délivre les données de paquet de train aux différents côtés application correspondants pour chaque canal.

10 30. Circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone et d'un paquet isochrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, caractérisé en ce qu'il comprend :

15 un circuit de traitement de données (120, ADP) servant à ajouter une étiquette auto-indicatrice aux données lues pour produire un paquet d'émission, et d'émettre celui-ci à destination du bus d'interface série, et, lors du transfert de données de l'autre noeud à son propre noeud, de produire un paquet de demande auquel il a ajouté une étiquette auto-indicatrice et d'émettre celui-ci à destination dudit bus d'interface série, de recevoir un paquet de réponse relativement au paquet de demande de la part de l'autre noeud, d'extraire la partie de données du

20 un circuit de commande (107) servant à recevoir un paquet de commande de la part de l'autre noeud et à faire démarrer le fonctionnement du circuit de traitement de données lorsque le contenu du paquet de commande indique une demande de transfert de données de son propre noeud à l'autre noeud et indique une demande de transfert de données de l'autre noeud à son propre

25 noeud ; et

30 un circuit de démultiplexage (105, 105a) servant à recevoir un paquet transféré depuis l'autre noeud via le bus d'interface série, à délivrer celui-ci au circuit de commande lorsque le paquet reçu est un paquet de commande, à délivrer celui-ci au circuit de traitement de données lorsque le paquet reçu est un paquet de réponse relativement à un paquet d'émission, et à délivrer celui-ci au côté application lorsque le paquet reçu est fait de données de paquet de train isochrone.

35 31. Circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet, auquel une information de destination de fourniture a été ajoutée, entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, caractérisé en ce qu'il comprend :

un circuit de démultiplexage (105, 105a) servant à demultiplexer un paquet reçu sur la base de l'information de destination de fourniture du paquet reçu et à délivrer celui-ci à la destination de fourniture correspondante.

5 32. Circuit d'interface série selon la revendication 31, caractérisé en ce que le paquet reçu est un paquet asynchrone, et l'information de destination de fourniture est une information d'étiquette de transaction et de code de transaction.

33. Circuit d'interface série selon la revendication 31, caractérisé en ce que le paquet reçu est un paquet isochrone, et l'information de destination de fourniture est une information de code de transaction et de canal.

10 34. Procédé de traitement de signaux d'un circuit d'interface série permettant d'effectuer l'émission et la réception d'un paquet asynchrone entre son propre noeud et un autre noeud connecté à son propre noeud via un bus d'interface série, ledit procédé de traitement de signaux du circuit d'interface série étant caractérisé en ce qu'il comprend, lors du transfert de données de l'autre noeud à
15 son propre noeud, les opérations suivantes :

produire un paquet de demande auquel il a ajouté une étiquette auto-indicatrice et émettre celui-ci à destination du bus d'interface série,

recevoir un paquet de réponse relativement à ce paquet de demande de la part de l'autre noeud, et

20 extraire la partie de données du paquet de réponse.

35. Procédé de traitement de signaux selon la revendication 34, caractérisé en ce qu'il comprend en outre les opérations consistant à produire une pluralité de paquets de demande indiqués à l'aide d'étiquettes et à émettre ceux-ci à destination du bus d'interface série de façon à permettre le transfert de données en
25 paquets.

36. Procédé de traitement de signaux selon la revendication 34 ou 35, caractérisé en ce qu'il comprend en outre, lors de la réception d'un paquet de réponse, l'opération consistant à extraire le paquet de réception de données relativement à son propre noeud sur la base de l'étiquette ajoutée au paquet de
30 réponse et d'un code de réponse prédéterminé.

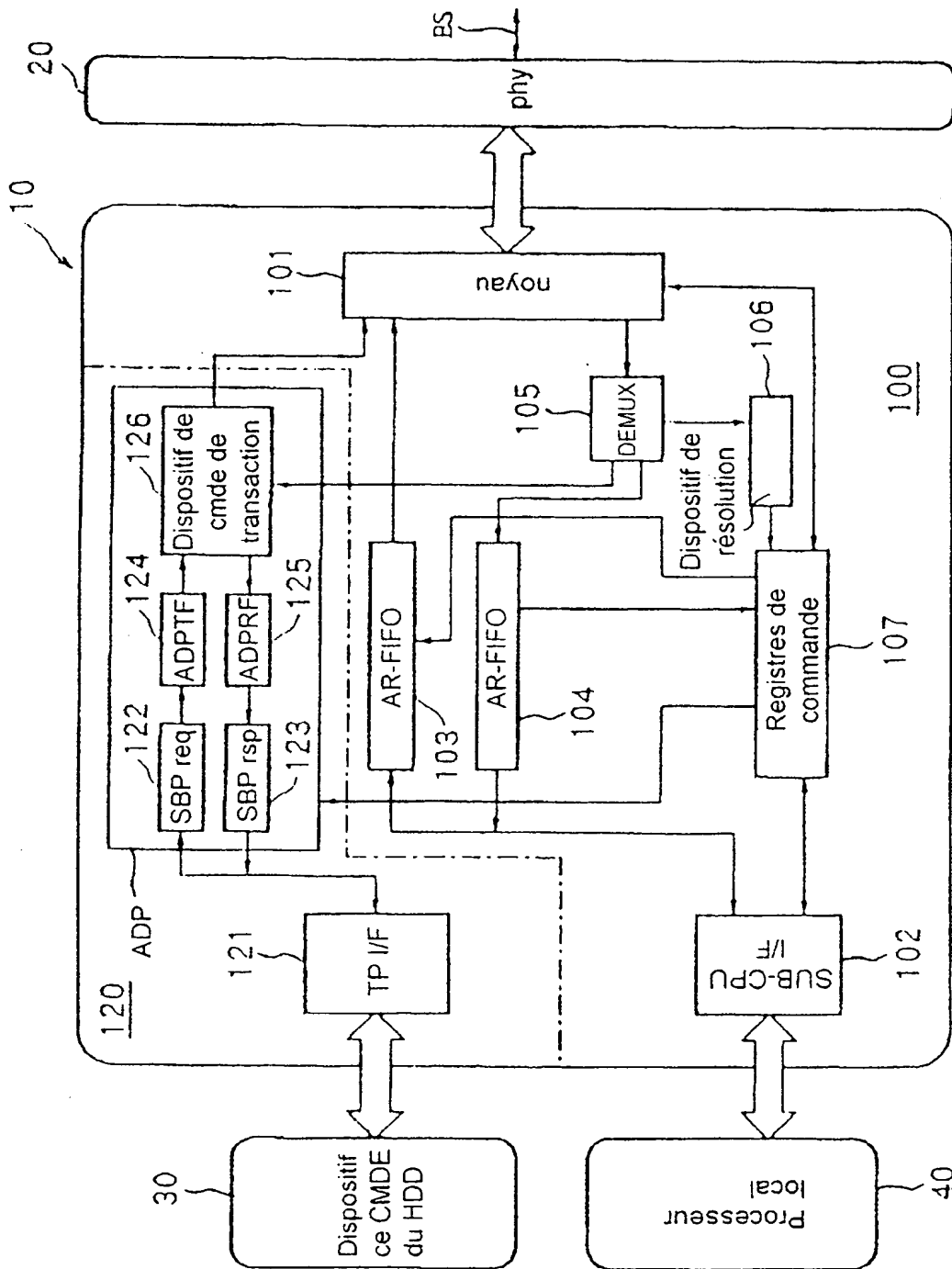


FIG.1

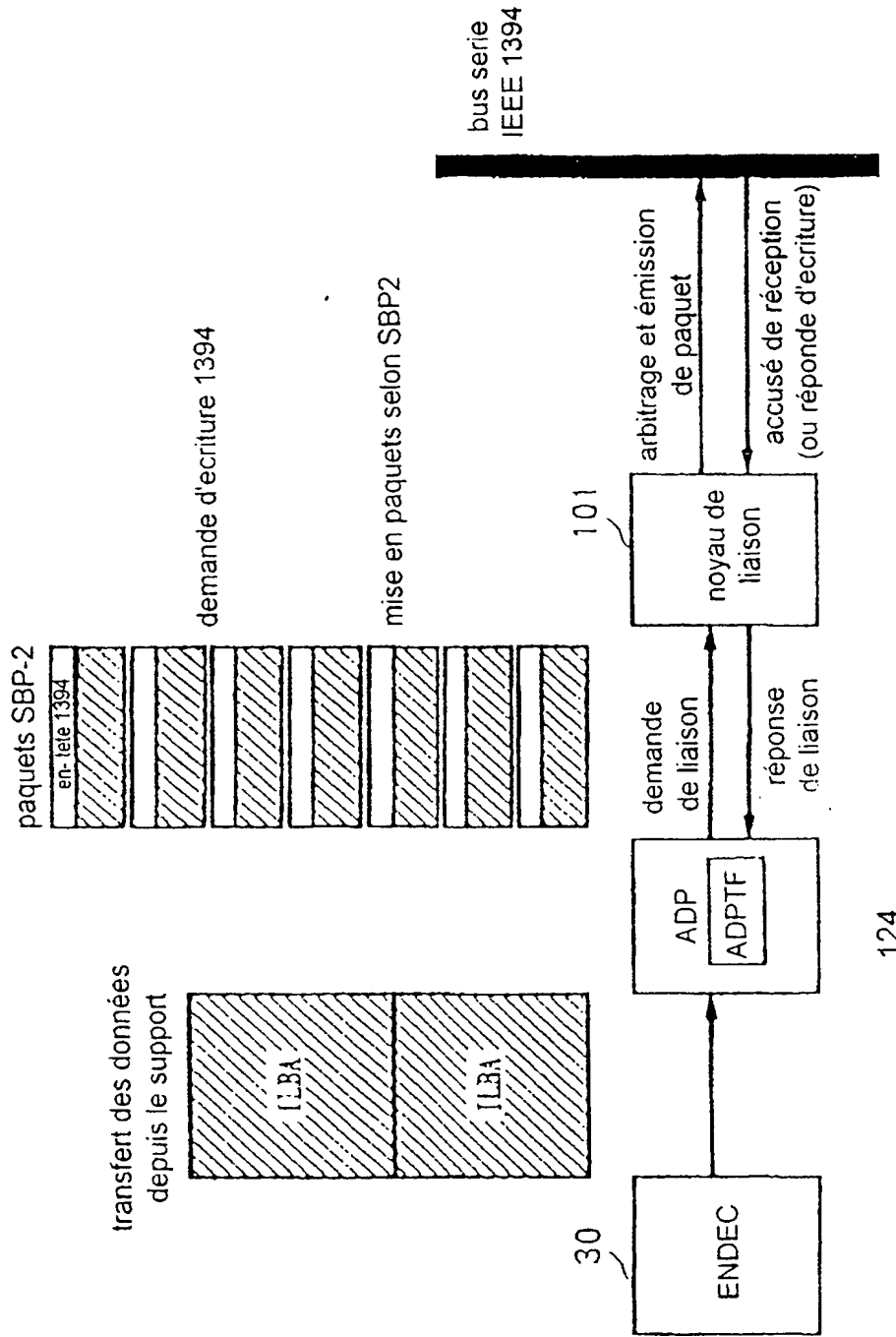


FIG.2

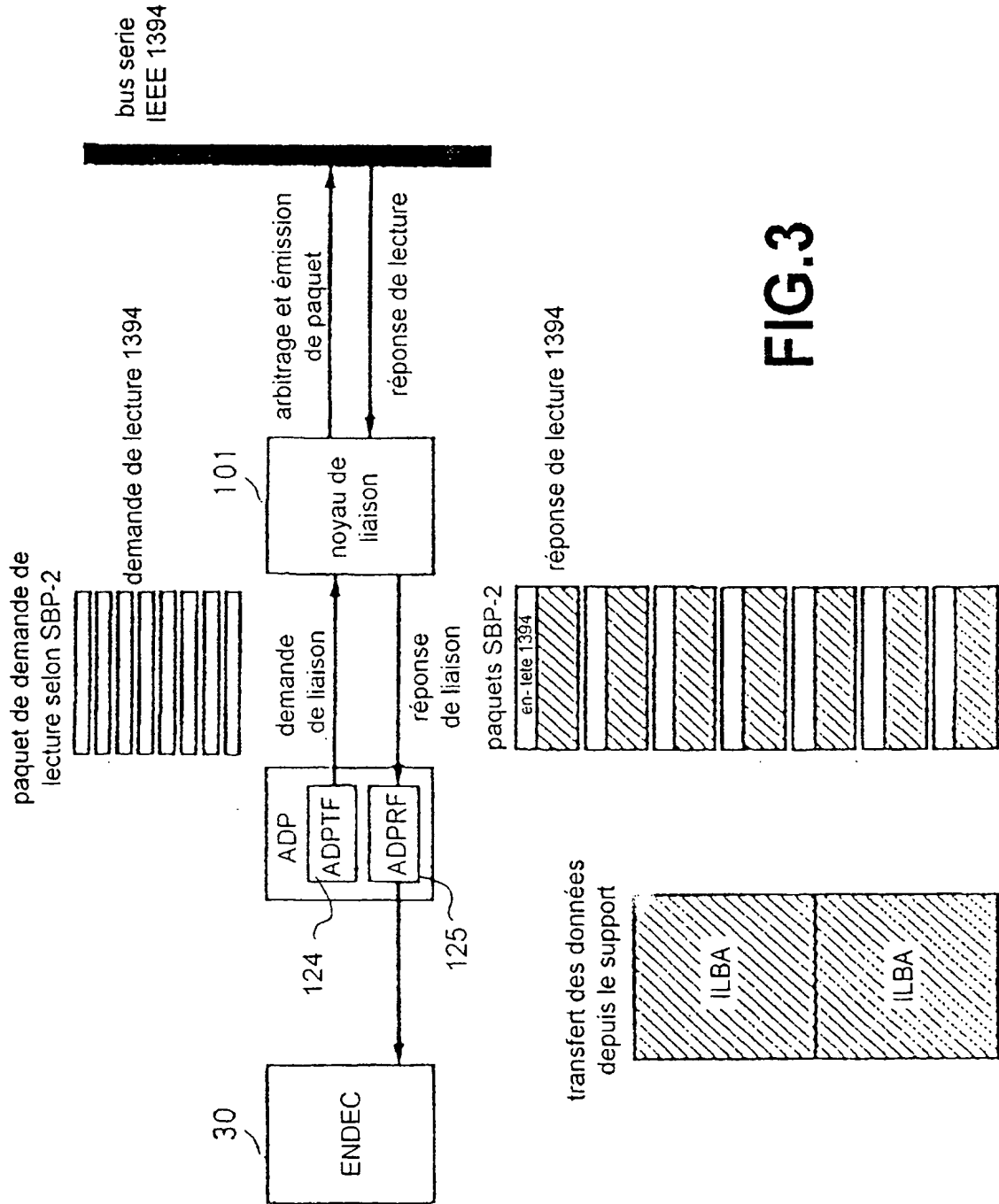


FIG.3

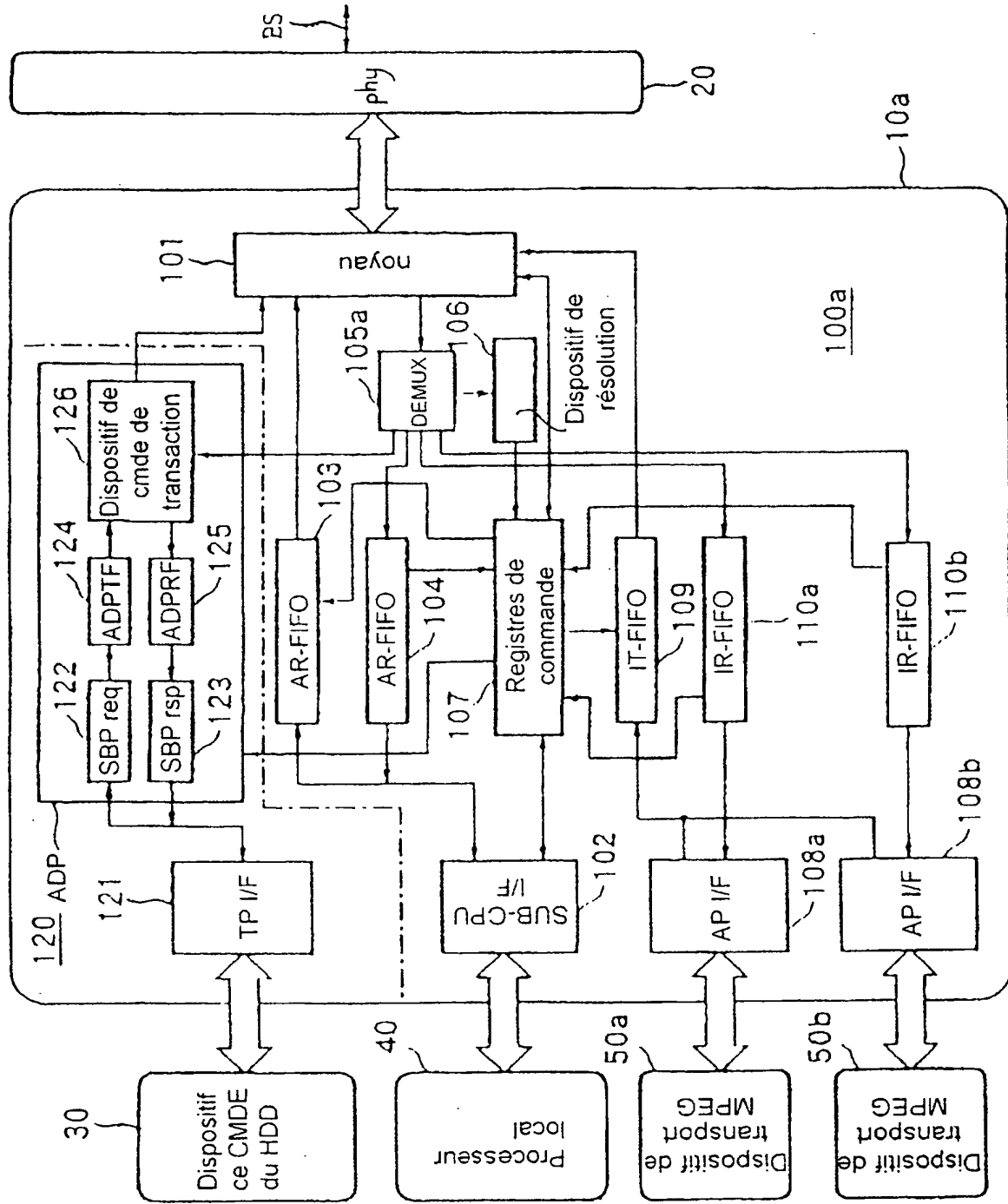


FIG.4

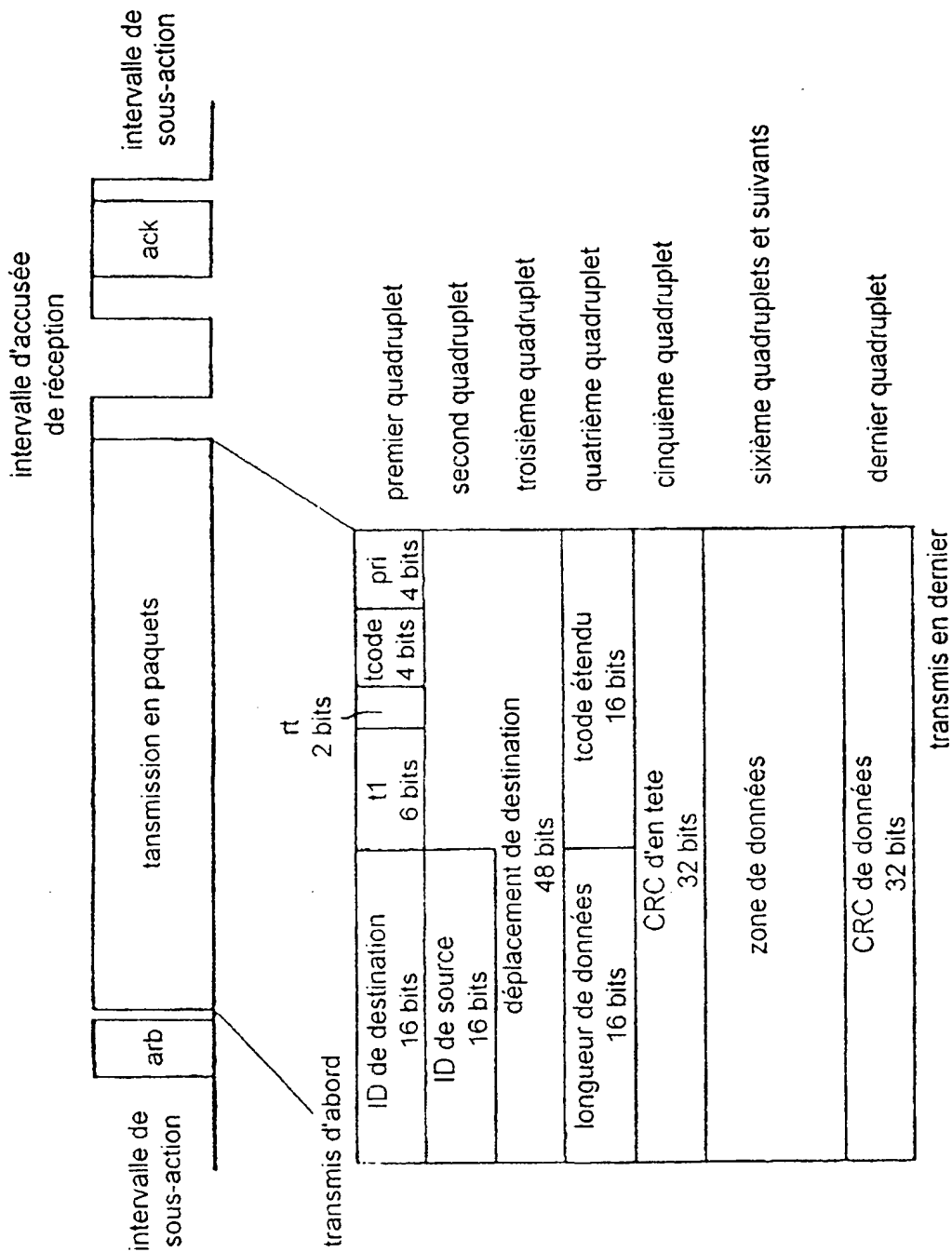
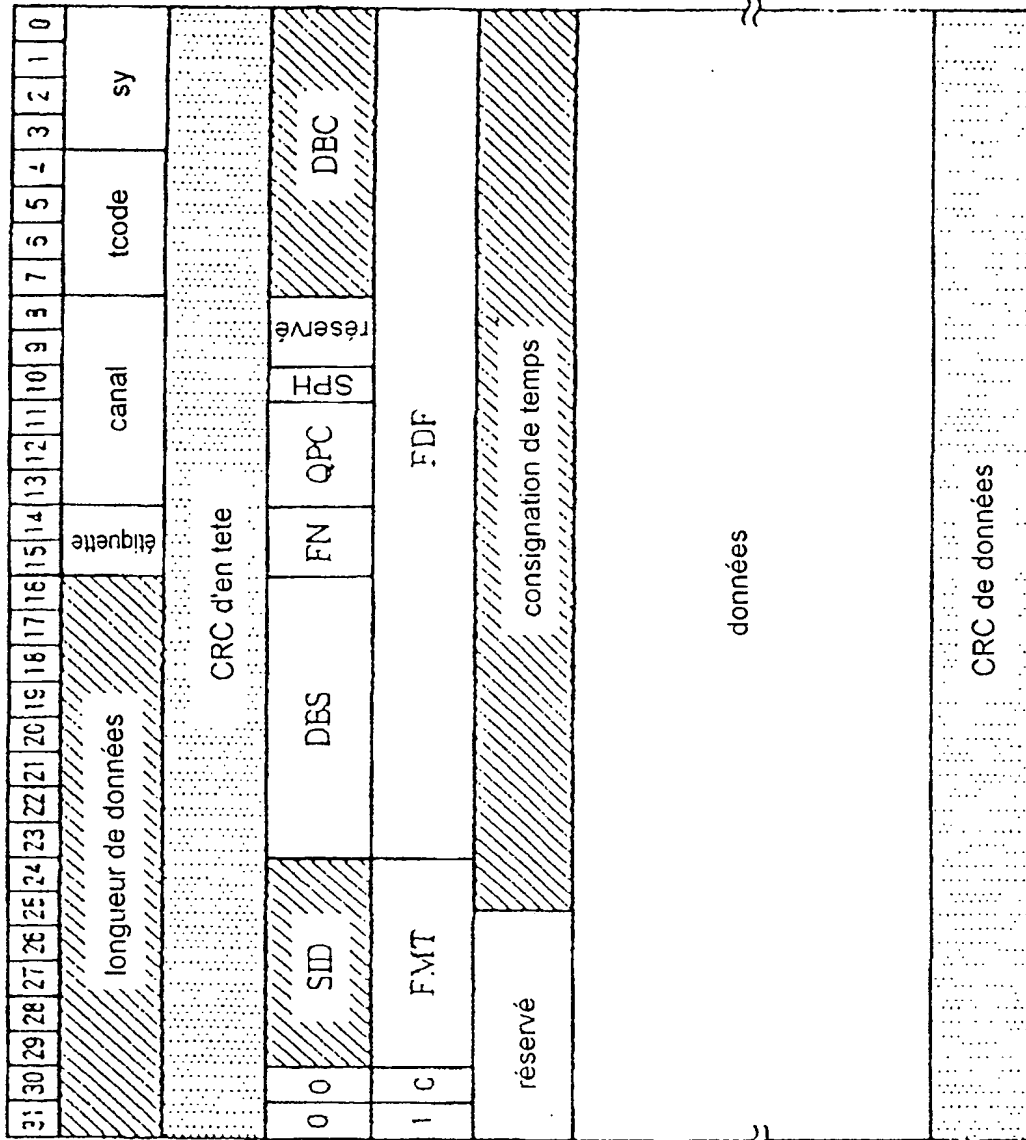


FIG. 5A

FIG. 5B



premier quadruplet : en-tête 1394

deuxième quadruplet

troisième quadruplet : en-tête 1 de CIP

quatrième quadruplet : en-tête 2 de CIP

cinquième quadruplet : SPH

sixième quadruplet et suivants

dernier quadruplet

FIG.6